SAP® MaxDB™ Expert Session

SAP® MaxDB™: Introduction into Remote SQL Server

Heike Gursch August 27, 2013

Public

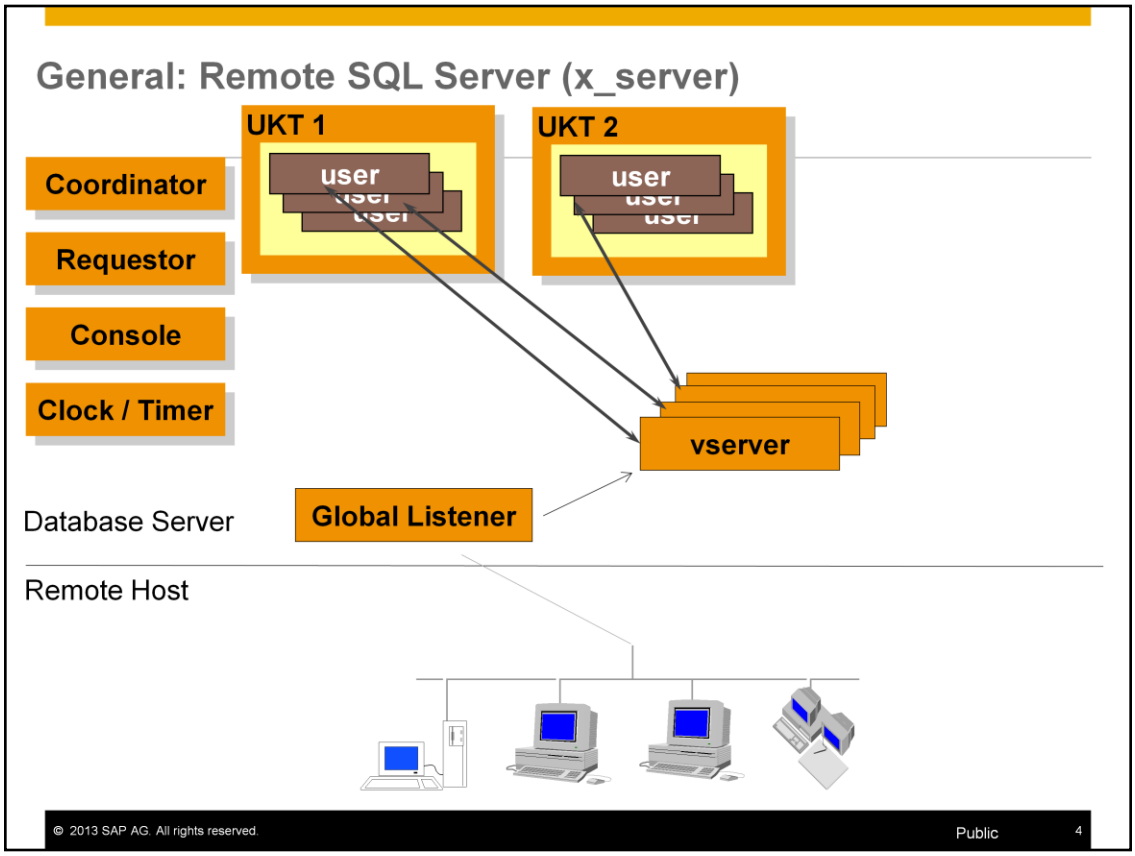
SAP® MaxDB™ – Expert Session

Introduction into SAP ® MaxDB™ Remote SQL server

Heike Gursch
Christiane Hienger
IMS MaxDB/liveCache Development Support
August 27, 2013

Agenda

- General: Remote SQL Server (x_server)
- Start phase (non-isolated)
- Global Listener (general and start phase)
- Remote connect for different constellations
- Start and Stop of the Remote SQL Server
- More options
- Port Numbers (isolated installation)
- Ports and Network Protocols
- Process list
- Logfiles of the Remote SQL Server
- Upgrade of a MaxDB version



The **X-Server** is the communication server of the database system. It listens at a service port for connection requests from database applications and database tools. In the process list, this process is called **vserver**. A new vserver process is generated for every user process that logs on to the database remotely. The generating process serves the user; the new process waits for the next user logon. On Windows, an additional thread is started for the user logon.

On Windows, the X-Server runs as a service.

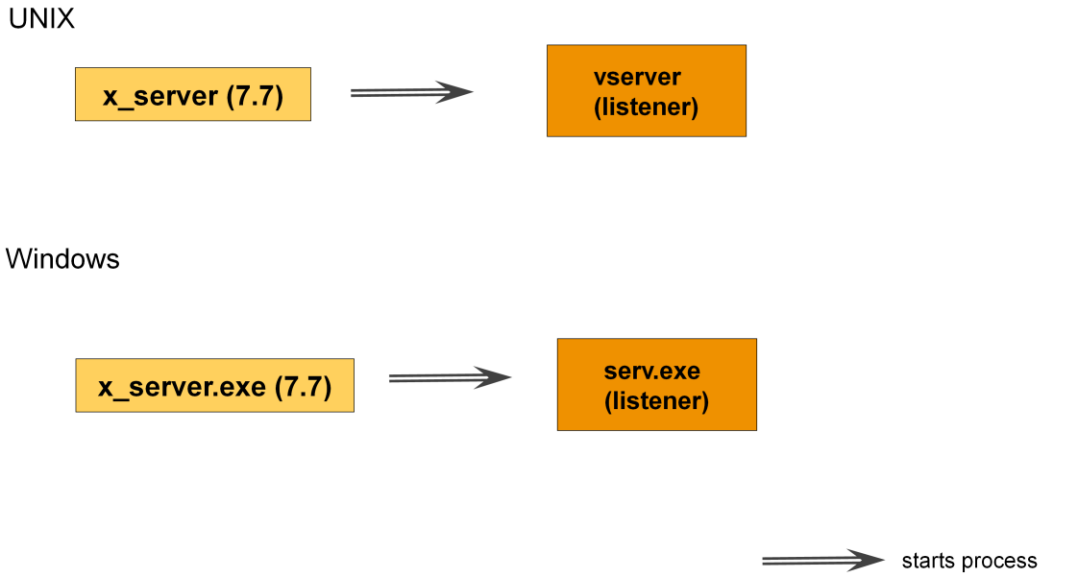
Local user sessions communicate with the database instance via a shared memory.

If you want to force the database system to establish the connection via an X server, specify localhost as the database computer when logging on to the database.

Note that the JDBC interface needs the x_server even if it is installed on the same computer as the database.

MaxDB 7.8 introduced the isolated software installation. Every database installed for SAP application uses its own port number. Clients first connect to a global listener which returns the instance specific port number. The client then connects to the x_server assigned to the instance.

x_server (start phase) – non-isolated (<= 7.7)



Firstly we have a look at the start phase of the x_server in an environment with databases of versions up to 7.7. This is a non-isolated software environment.

Under UNIX the x_server program starts a process with the name vserver which acts as listener.

Under Windows a thread serv.exe is started which also acts as a listener.

These pictures just describe the initial start. No connect from a remote client has taken place so far.

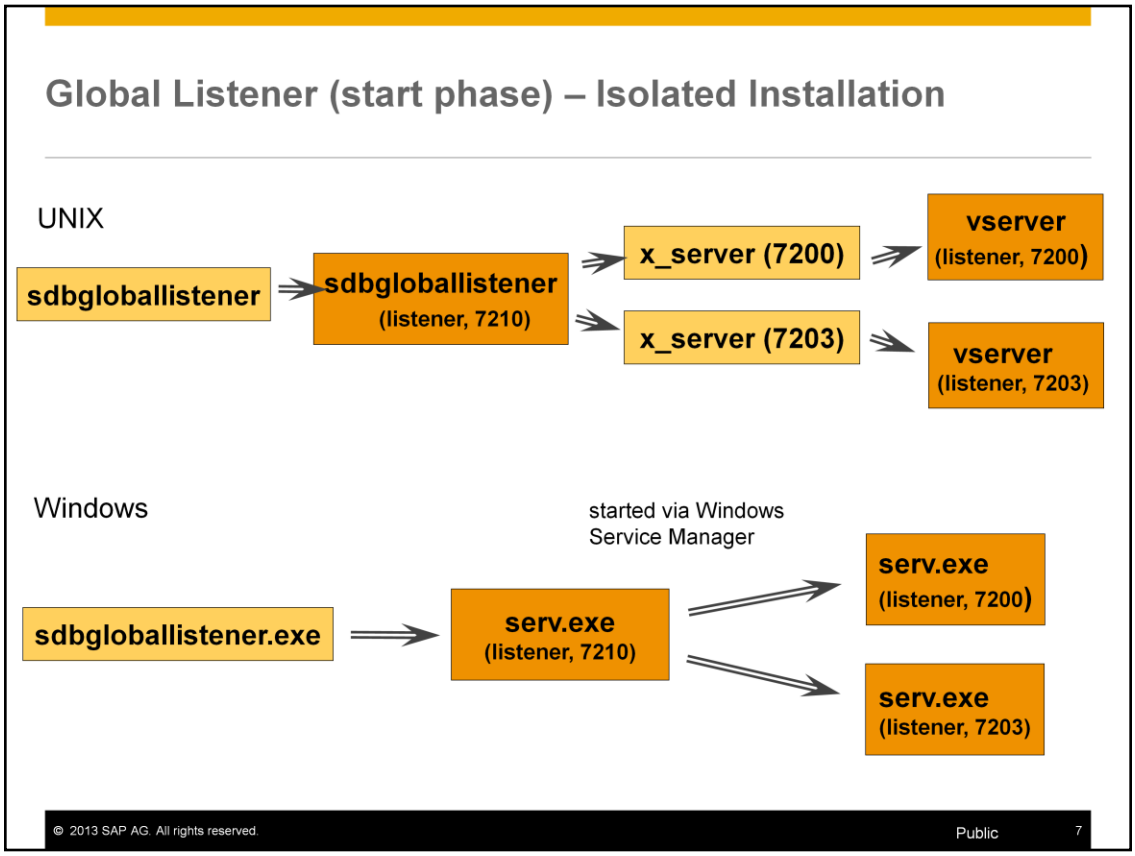
Global Listener

- was introduced with the isolated installation (as of 7.8)
- not only one x_server for all installed databases on a host
- an x_server with a separate port number is assigned to each database installation

- the global listener is a special x_server

- during logon (connect) the remote client contacts a global listener
- the global listener communicates to the remote client the port number of the database-specific x_server (assigned to the installation)
- the remote client automatically connects to the x_server that belongs to the installation

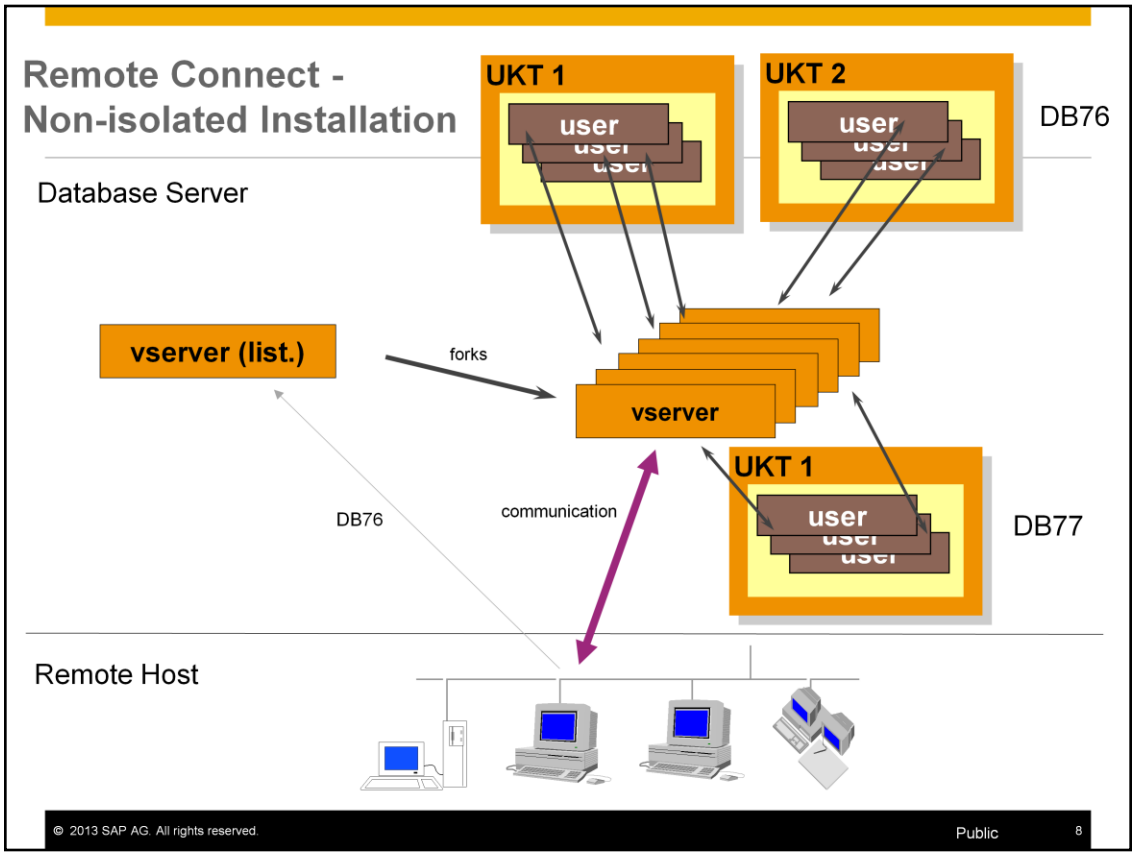
Before it is shown how the start phase works in an isolated environment the concept of the global listener is introduced.



These pictures show the start phase of the global listener and its depending processes. No connect has taken place so far.

Under UNIX the sdbgloballistener starts a process sdbgloballistener which listens to a specific port number. For each database the corresponding x_server program is started which starts a first vserver process which acts as listener.

In principle it‘s the same for Windows. The names are different. You find a serv.exe in the list of services which acts as global listener. Each database is served by its own serv.exe (acting as listener) which is started directly by the Windows Service Manager.



The following slides show how a connect is handled that has been issued from a remote host.

This picture shows how a connect is done (in a UNIX system) within a non-isolated environment. There are databases of versions 7.6 and 7.7.

The vserver(listener) must be of the highest database version that is installed on the server.

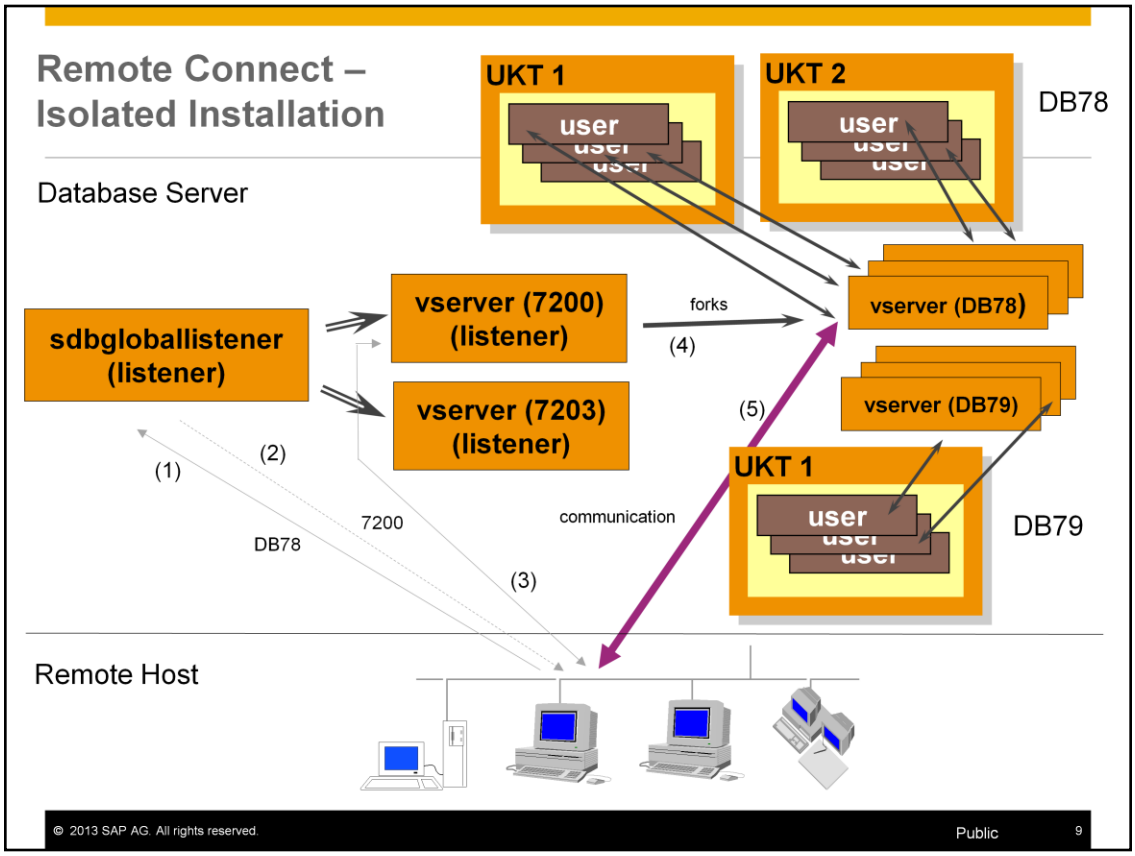
In this example it has to be updated if a higher patch of 7.7 is installed.

Example:

The client sends a connect request to DB76 to the vserver(listener) running on the database host.

The vserver(listener) forks a process vserver. A user task of one of the User Kernel Processes (UKT) of DB76 is assigned to this vserver.

From now on there is a direct communication between the client and its assigned vserver process.



To connect to a remote database, you have the following options:

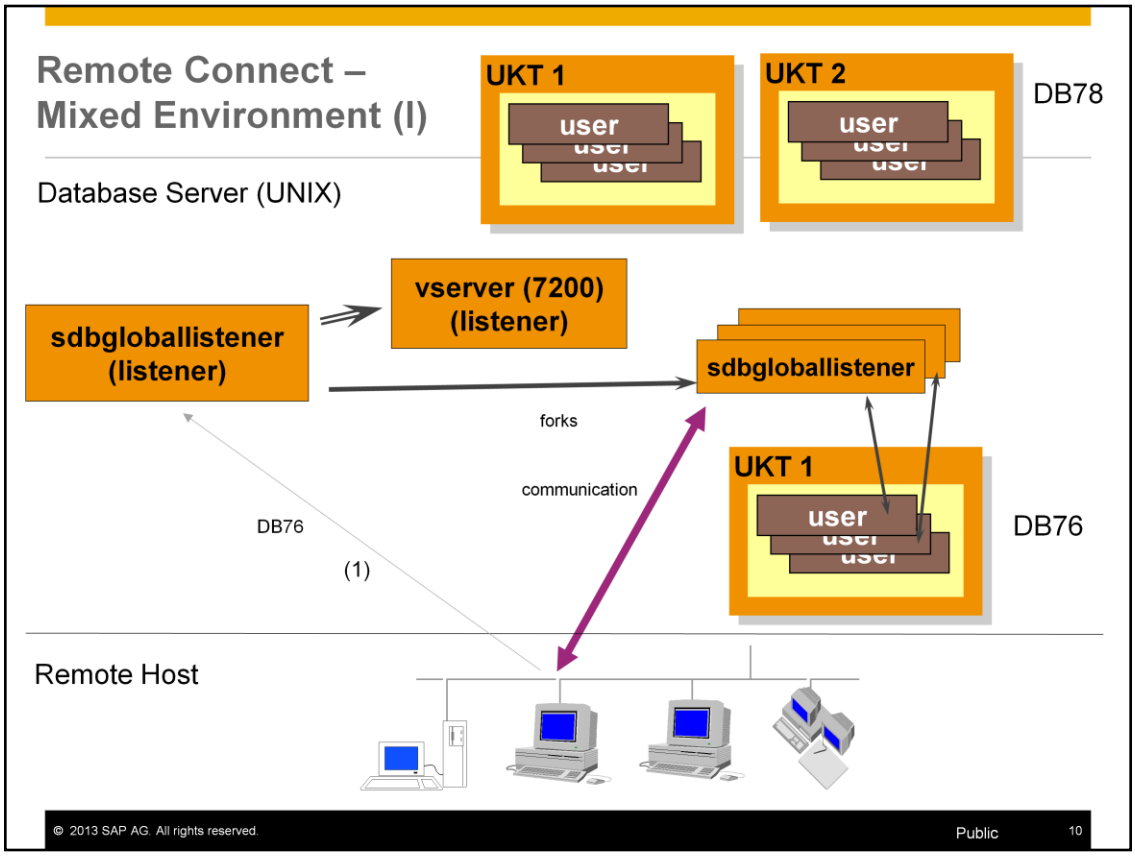
You specify the database name, the port number for the database and the database computer.

You are then directly connected to the vserver of the installation to which the database belongs to, and this vserver connects you to the database.

You only specify the database name and the database computer. Internally, the system executes the following steps:

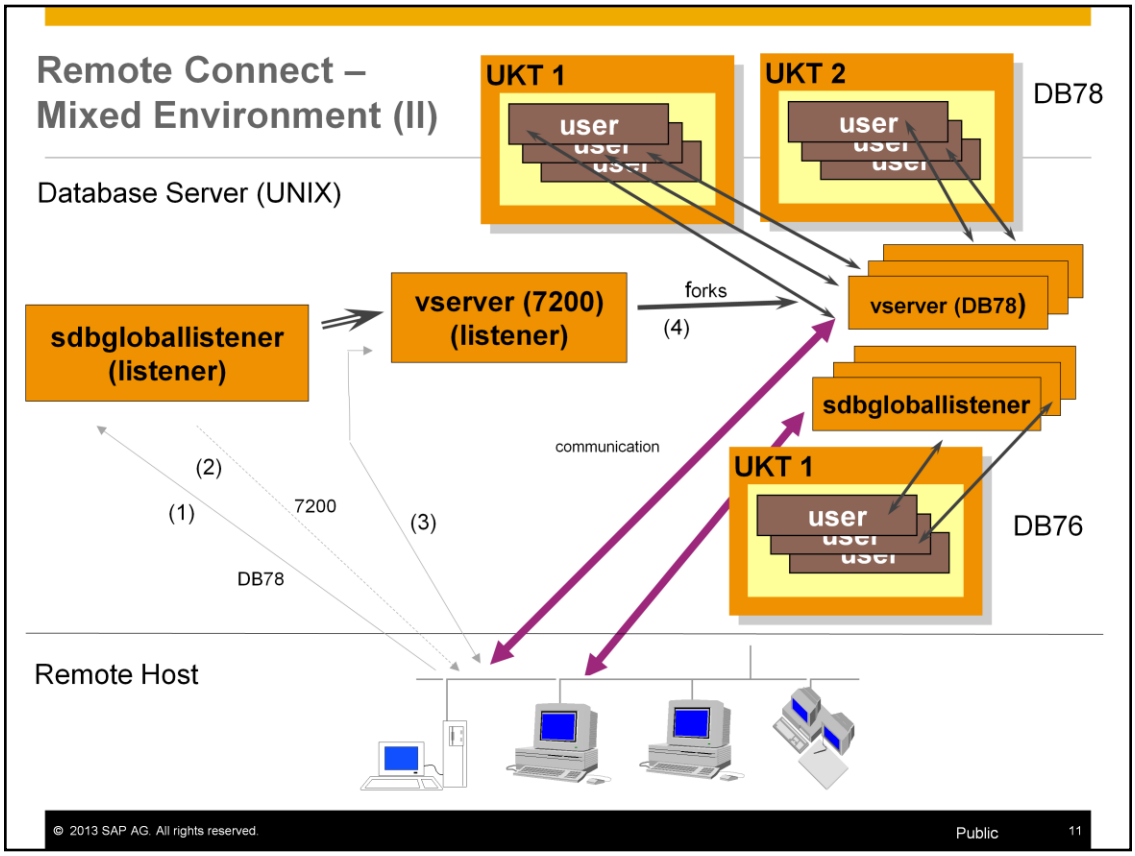
- during logon (connect) the remote client contacts the global listener (1)
- the global listener returns the port number of the database-specific vserver (assigned to the installation) to the remote client
  (port mapping) (2)
- the remote client automatically connects to the vserver that belongs to the installation (3)
- the vserver (listener) forks a vserver process (4)
- from now on there is direct communication between the remote client and the forked vserver process (5)

Vserver processes of DB78 are located in a different path than vservers of DB79. The database name is part of the path name.



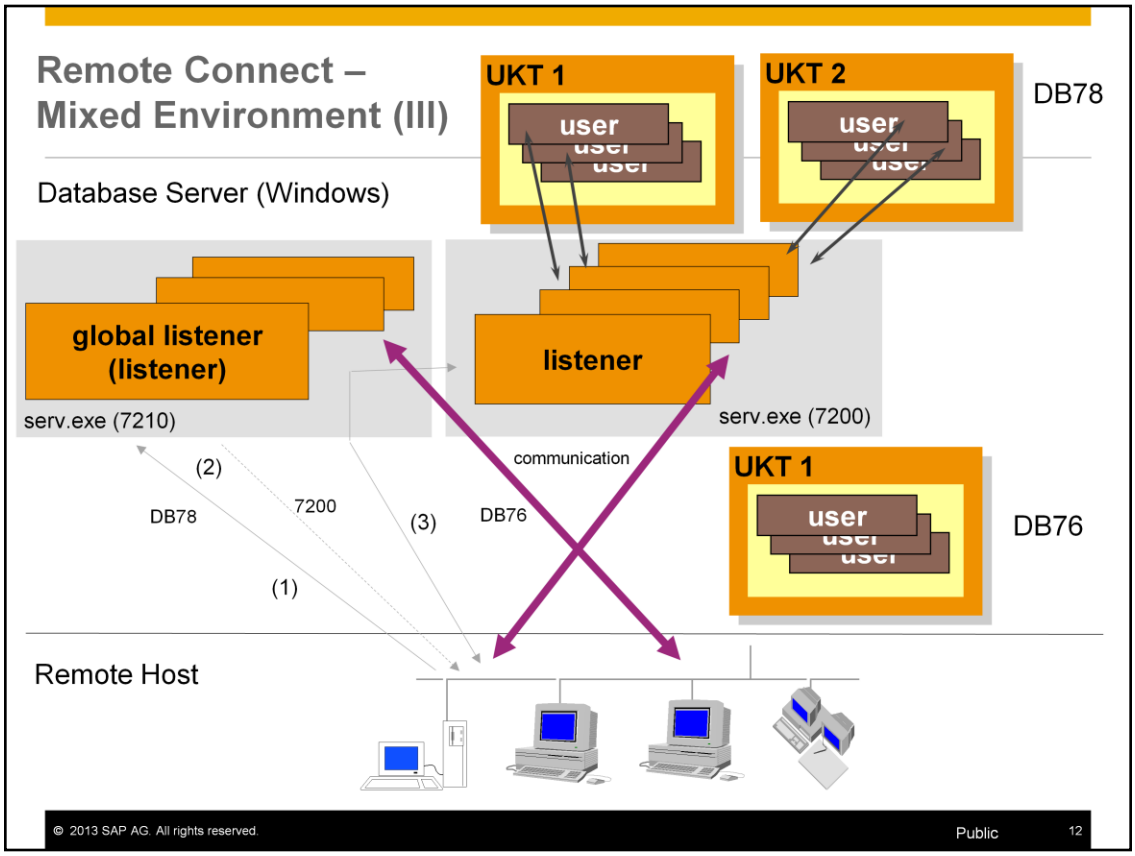
For the communication between older clients and databases as of version 7.8 as well as for communication between clients as of version 7.8 and older databases, the global listener acts as vserver (downward-compatibility). Only for communication between clients and databases that are both of version ≥ 7.8, the global listener additionally provides the port-mapping function as described before.

If a client connects to a database of version 7.6 then it also contacts the global listener and provides the name (and version) of the database. But as there is no special vserver assigned to this version the global listener acts as vserver itself and forks a process named sdbgloballistener. It does not contain the database name within its path name.



If additionally a (7.8.) client connects to a database of version 7.8 or higher the proceeding is the same as in a pure isolated environment:

- during logon (connect) the remote client contacts a global listener (1)
- the global listener returns the port number of the database-specific vserver (assigned to the installation) to the remote client (2)
- the remote client automatically connects to the vserver that belongs to the installation (3)
- the vserver (listener) forks a vserver process (4)
- from now on there is direct communication between the remote client and the vserver process (5)



In principle, in Windows environments the proceeding in a mixed environment with isolated and non-isolated installations is the same as in UNIX systems.

- during logon (connect) the remote client contacts a global listener (1)
- the global listener returns the port number of the database-specific vserver (assigned to the installation) to the remote client (2)
- the remote client automatically connects to the vserver that belongs to the installation (3)
- a thread within the serv.exe process is created
- from now on there is direct communication between the remote client and the newly created thread

The names of the processes differ and under Windows there is a thread concept.

The serv.exe (listener) with port number 7200 will not fork other processes but will create threads within the process serv.exe for the new connect request. The path containing the database name can be seen in the process list.

For connect requests for versions <= 7.7 the serv.exe with port number 7210 acts as global listener and will directly create new threads within its own process. There is no direct assignment to the database; the name is not contained in the process list.

Start and Stop of the Remote SQL Server (I)

Non-isolated installations (versions up to 7.7) – UNIX

- x_server is used by all databases on this host
- Version of the x_server is the highest version of database software
- In SAP environment: user of the group sdba (f.e. sqd<sid>)
- In SAP environment: use start scripts startdb and startsap (calls startdb) to start the x_server
- x_server passes on its environment and also the „ulimits“ to the database kernel

x_server start

x_server stop

Start and Stop of the Remote SQL Server (II)

Non-isolated installations (up to 7.7) – Windows

- x_server is used by all databases on this host
- Version of the x_server is the highest version of database software
- x_server runs as a service
- start type should be set to „Automatic“

x_server start

x_server stop

Start and Stop of the Remote SQL Server (III)

Isolated installations – UNIX

- In SAP environment: user of the group sdba (f.e. sqd<sid>)
- In SAP environment: use start scripts startdb and startsap (calls startdb) to start the global listener and the x_servers
- x_server passes on its environment and also the „ulimits“ to the database kernel
- sdbgloballistener and x_server for each database can be started and stopped independent of each other

```
<global_programs_path>\bin\sdbgloballistener start
<global_programs_path>\bin\sdbgloballistener [-all] stop
```

```
<installation_path>\bin\x_server  –S <tcp_port> start
<installation_path>\bin\x_server  [-S <tcp_port>] stop
```


Usually with the start of the sdbgloballistener also the x_servers for the different installations are started.

(As of 7.9 you can change that behaviour by setting a special „automatic flag“ during x_server installation. For example in FlexFrame environments it might make sense to prevent the automatic start process for all x_servers.)

If the global listener is already running and several databases have been installed afterwards, the x_servers for those databases have to be started explicitly.

If the option –all is specified when stopping the global listener, all database-dependent x_servers are also stopped.

Without option –all only the global listener is stopped and all x_servers can be stopped individually.

The different pathes can be identified with

dbmcli –u <dbm>,<pw> dbm_getpath

Example output:

ClientProgPath=/sapdb/DB78/db

InstallationPath=/sapdb/DB78/db

GlobalProgPath=/sapdb/programs

DataPath=/sapdb/DB78/data

GlobalDataPath=/sapdb/data

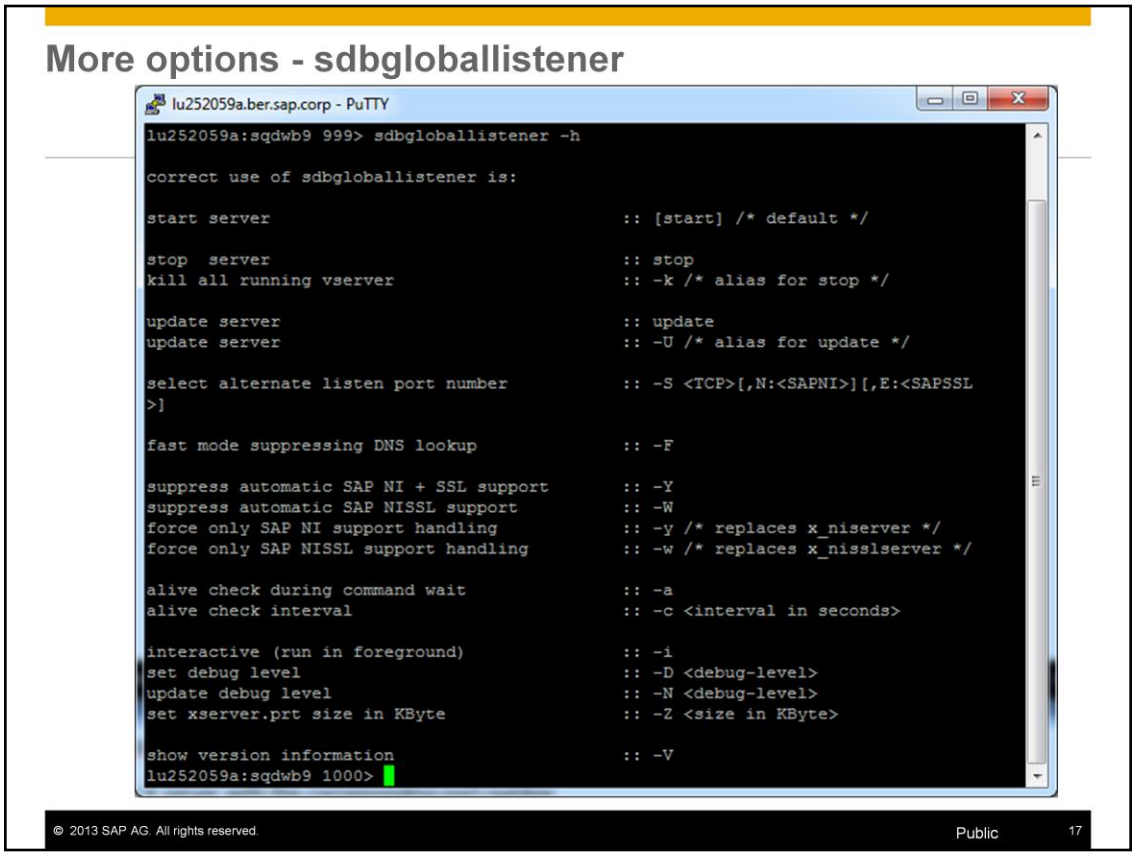
Start and Stop of the Remote SQL Server (IV)

Isolated installations – Windows

- global listener (x_server with port number 7210) and the database specific x_servers run as a service
- start type should be set to „Automatic“

x_server start

x_server stop



-S <tcp_port>[,N:<ni_port][,E:<ssl_port>]

Only starts the installation-specific X server with the corresponding port number.

Note that to specify NI and SSL port numbers, you must use the sdbglobalallistener program (not the x_server program).

-F

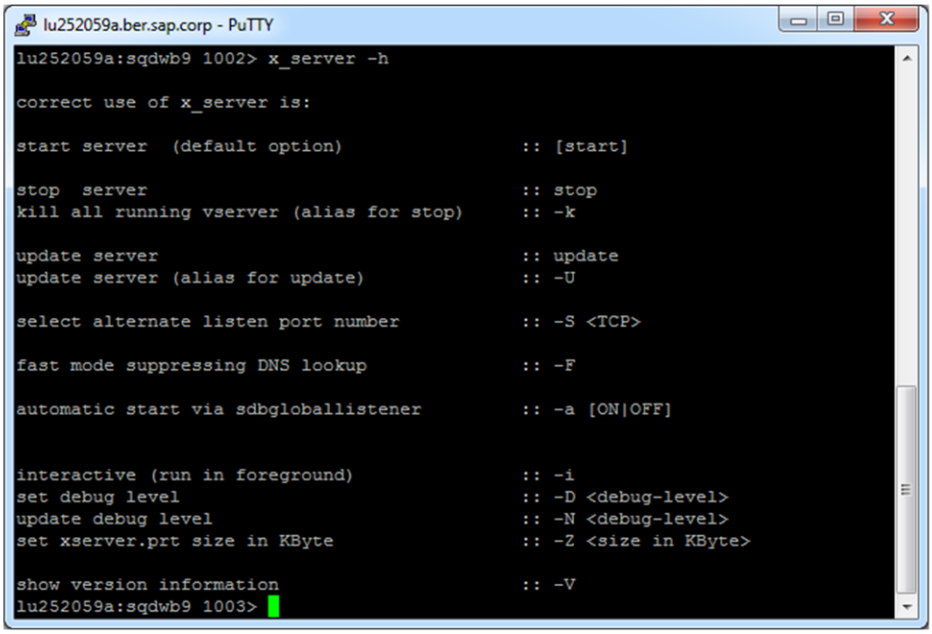
If you set this option, the program does not determine the computer name of the application logging on to the DNS server (no reverse DNS lookup).

If it is taking a long time to connect to a remote database, restart the program with this option. Like this, you can find out whether the DNS server is responsible for the slow connection.

-U (UNIX only)

When the sdbglobalallistener has to be exchanged it is not necessary to stop the whole environment (all x_servers of running databases). The –U option can be used to perform the upgrade in the background.

More options – x_server



-S <tcp_port>

Only starts the installation-specific x_server with the corresponding port number

-F

If you set this option, the program does not determine the computer name of the application logging on to the DNS server (no reverse DNS lookup).

If it is taking a long time to connect to a remote database, restart program with this option. Like this, you can find out whether the DNS server is responsible for the slow connection.

-a [ON|OFF]

Enables/disables the automatic start of the x_server by the sdbglobalistener

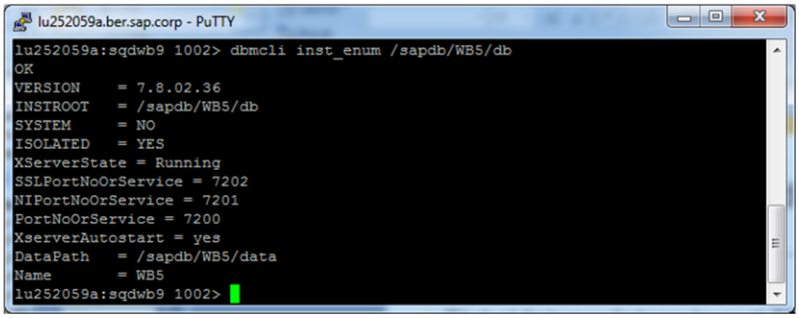
-U (UNIX only)

When the x_server has to be exchanged it is not necessary to stop the whole environment. The –U option can be used to perform the upgrade in the background.

Port Numbers (Isolated Installation)

- Global listener listens to port 7210
- x_server of the first database listens to port 7200, the second to 7203 and so on
- Find out which port is used:

dbmcli inst_enum <InstallationPath>



There are default numbers that are used as port number.

The global listener usually listens to port number 7210.

Other port numbers could be used by making changes in /etc/services.

The port numbers for the different databases are incremented by 3.

You see in the example that per database three different port numbers are used. The additional ones are used for NI connections and for encryption purposes. As port number 7210 is usually reserved for the global listener the installation tool will look for the next free port number.

If a database is dropped and the port numbers are free again, they will be used for newly installed databases.

In future there will be an additional option for inst_enum; alternatively the database name can be specified:

dbmcli inst_enum <DB name>

Special port numbers (I)

Ports and Protocols of the SAP MaxDB X Servers

Scope	Default Port	Function of the X Server	Protocol	Protocol Identifier
All installations on the database computer	7210	Global listener	TCP/IP	`remote://`
	7269	Global listener with SAP networtk protocol NI (for connections via SAPRouter, only available in SAP systems)	NI (based on TCP/IP)	`sapni://`
	7270	Global listener with SAP networtk protocol NI and SAP encryption library (for connections via SAPRouter, only available in SAP systems)	NISSL (based on SSL/TLS)	`remotes://` `sapnis://`

The protocol identifier can be used to determine the URI (Uniform Resource Identifier).

The URI is a compact string of characters used to identify MaxDB/liveCache specific resources.

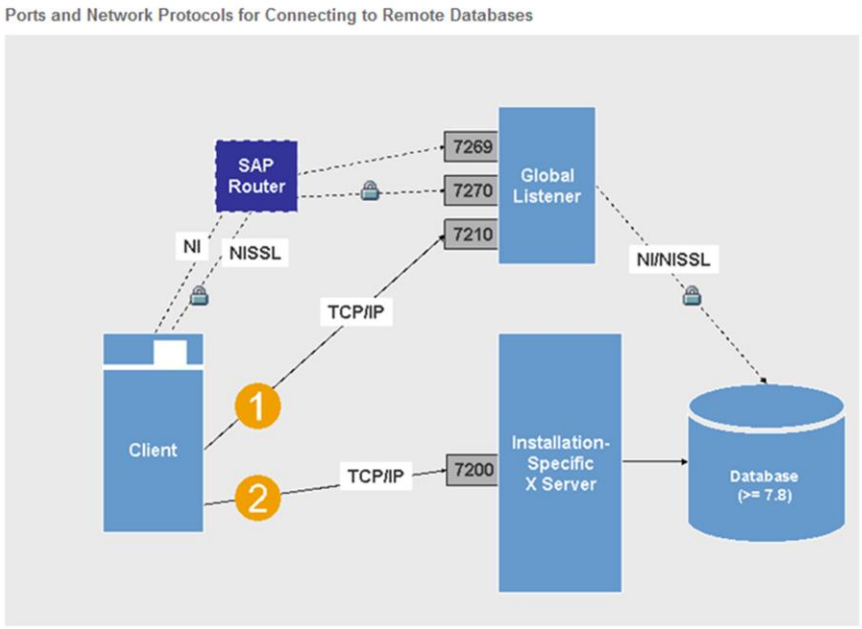
Any URI syntax is dependent on a specific scheme which will not be explained here in detail. To get an impression see the following example:

liveCache:remote://mypc:3322/database/myserverdb/procserver/12345678

Special port numbers (II)

Scope	Default Port	Function of the X Server	Protocol	Protocol Identifier
First installation <installation_1> on the database computer	7200	X server for <installation_1>	TCP/IP	remote://
Second installation <installation_2> on the database computer	7203	X server for <installation_2>	TCP/IP	remote://

Overview: Ports and Network Protocols



Connecting via SAPRouter

Connections to the database via SAPRouter are always established via the global listener. These connections use the NI or the NISSL protocol with designated ports.

To encrypt the data transfer between the client and the global listener/X servers, SAP customers can use SSL/TLS.

Process List (Isolated Installation)

On UNIX:

- sdbgloballistener and vserver processes
- For each remote logon (connect) a separate vserver process is started
- Use ps command for identification

```
sdb       7328     1  0 Jan15 ?        00:00:00 sdbgloballistener

sdb 7316    .../sapdb/EXPERTDB/db/pgm/vserver -sdbstarter -S 7203 -Y
sdb 7318  7316 /sapdb/EXPERTDB/db/pgm/vserver -sdbstarter -S 7203
```

On Windows:

- serv.exe in the list of services
- Multi-thread application

UNIX:

The output of the ps command (ps –ef | grep vserver) determines which vserver process communicates with which database via the path name. In the example the vserver processes belong to the database EXPERTDB. If there are several sdbgloballistener processes in the process list then you can conclude that databases with versions <= 7.7 are operated.

Windows:

Multi-thread application:

Only one process runs and a new thread is started for each remote logon (connect)

Which process serv.exe belongs to which database cannot be directly seen. You can activate the path display in the ‚Command line' and identify the database in the display of processes.

Logfiles of the Remote SQL Server

Global listener:

- Stored in <GlobalDataPath>/wrk
- > 7.8: sdbgloballistener_<host>_<port>.prt
- 7.8: sdbgloballistener_<host>.prt
- After stop and restart the old file is saved with name sdbglobalIlistener_<host>_<port>.old
  (In older versions you will also find the names xserver_<computer_name>.old or xserver_<computer_name>_<port>.old for this logfile.)

Database-dependent x_servers:

- Stored in <PrivateDataPath>/wrk
- Contain the port number in the log name

GlobalDataPath and PrivateDataPath can be determined by use of the xinstinfo tool.

The pathes can also be displayed with
dbmcli –u <dbm>,<pw> dbm_getpath

Display the logfiles

Database Studio:

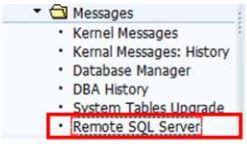
Diagnosis Files -> Extended File List (right click)

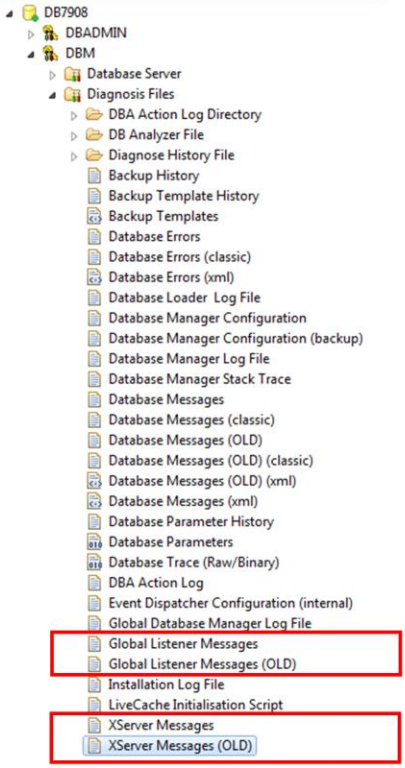
-> Xserver Messages

DBA Cockpit:

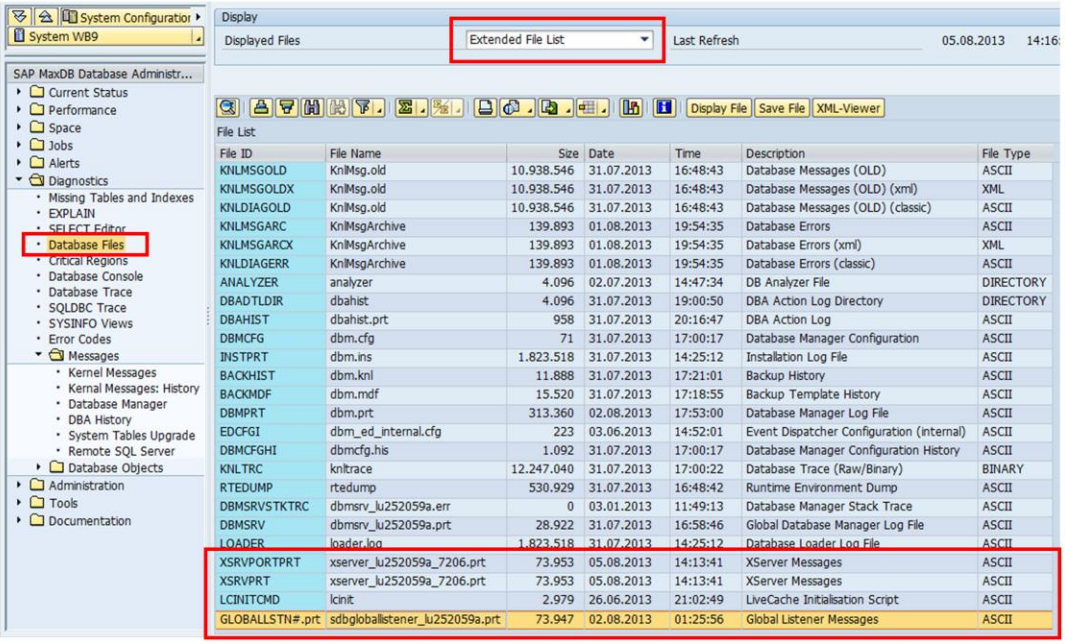
Diagnostics -> Messages -> Remote SQL server

Diagnostics -> Database Files -> Extended File List





Display the logfiles in DBACockpit



Content of logfiles

- Name and environment variables of the user who started the remote SQL servers
- Operating system settings that are passed on to the database
- Communication problems
- Network problems

Example of logfile (sdbglobalistener)

```
-------------------------------------------------------------------------------
Date       Time       PID Typ  MsgID Label    Message Text
-------------------------------------------------------------------------------
2013-07-31 15:54:54 77312 INF  12903          sdbglobalistener on port 7210 started
2013-07-31 15:54:54 77312 INF  12904 XSERVER  Service port is 7210
2013-07-31 15:54:54 77312 INF  12922 ENVIRON  Command line arguments
2013-07-31 15:54:54 77312 INF  12924 ENVIRON   [1] -m
2013-07-31 15:54:54 77312 INF  12923 ENVIRON  Command line argument dump completed
2013-07-31 15:54:54 77312 INF  12898 ENVIRON  Resource limit dump start
2013-07-31 15:54:54 77312 INF  12898 ENVIRON  Started by user id 1001 group id 1002
2013-07-31 15:54:54 77312 INF  12898 ENVIRON  Current user id 1001 effective id 1001
2013-07-31 15:54:54 77312 INF  12898 ENVIRON  Current group id 1002 effective id 1002
2013-07-31 15:54:54 77312 INF  12898 ENVIRON   cpu time unlimited
2013-07-31 15:54:54 77312 INF  12898 ENVIRON   number of processes 96101
2013-07-31 15:54:54 77312 INF  12898 ENVIRON   number of open files 32800
2013-07-31 15:54:54 77312 INF  12898 ENVIRON   core size 0 KBytes
2013-07-31 15:54:54 77312 INF  12898 ENVIRON   file size unlimited
2013-07-31 15:54:54 77312 INF  12898 ENVIRON   heap memory size unlimited
2013-07-31 15:54:54 77312 INF  12898 ENVIRON   stack memory size 8192 KBytes
2013-07-31 15:54:54 77312 INF  12898 ENVIRON   lockable memory size 64 KBytes
2013-07-31 15:54:54 77312 INF  12898 ENVIRON   virtual memory size unlimited
2013-07-31 15:54:54 77312 INF  12898 ENVIRON   resident set size unlimited
2013-07-31 15:54:54 77312 INF  12898 ENVIRON  Resource limit dump completed
2013-07-31 15:54:54 77312 INF  12898 ENVIRON  Environment dump start
2013-07-31 15:54:54 77312 INF  12898 ENVIRON   DBROOT=/sapdb/programs
2013-07-31 15:54:54 77312 INF  12898 ENVIRON   IFS=

2013-07-31 15:54:54 77312 INF  12898 ENVIRON   LD_LIBRARY_PATH=/sapdb/programs/lib
2013-07-31 15:54:54 77312 INF  12898 ENVIRON   LOGNAME=sdb
2013-07-31 15:54:54 77312 INF  12898 ENVIRON   PATH=/bin:/usr/bin
2013-07-31 15:54:54 77312 INF  12898 ENVIRON   TMP=/var/tmp
2013-07-31 15:54:54 77312 INF  12898 ENVIRON   TMPDIR=/var/tmp
2013-07-31 15:54:54 77312 INF  12898 ENVIRON   USER=sdb
2013-07-31 15:54:54 77312 INF  12898 ENVIRON  Environment dump completed
------------------------------ end of startup part ------------------------------
2013-07-31 15:54:54 77312 WNG  12458 XSERVER  NISSL Utility init: CDKLoadSAPCryptModule: 51
2013-07-31 21:25:02 77312 ERR  11926 XSERVER  Receive packet, Ref:6 - socket recv error (110:Connection timed out
2013-07-31 21:25:04 77312 ERR  11926 XSERVER  Receive packet, Ref:6 - socket recv error (110:Connection timed out
2013-08-01 23:26:50 77312 ERR  11926 XSERVER  Receive packet, Ref:6 - socket recv error (110:Connection timed out
```

Example of logfile (x_server)

```
------------------------------------------------------------------------------
Date       Time       PID   TID Typ  MsgID Label    Message Text
------------------------------------------------------------------------------
2013-07-31 15:54:52  7088 83264 INF  12903          XServer on port 7206 started
2013-07-31 15:54:52  7088 83264 INF  13010            installation WB9  - path: /sapdb/WB9/db
2013-07-31 15:54:52  7088 83264 INF  12904 XSERVER  Service port is 7206
2013-07-31 15:54:52  7088 83264 INF  12922 ENVIRON  Command line arguments
2013-07-31 15:54:52  7088 83264 INF  12924 ENVIRON   [1] -S
2013-07-31 15:54:52  7088 83264 INF  12924 ENVIRON   [2] 7206
2013-07-31 15:54:52  7088 83264 INF  12924 ENVIRON   [3] -Y
2013-07-31 15:54:52  7088 83264 INF  12923 ENVIRON  Command line argument dump completed
2013-07-31 15:54:52  7088 83264 INF  12898 ENVIRON  Resource limit dump start
2013-07-31 15:54:52  7088 83264 INF  12898 ENVIRON  Started by user id 1001 group id 1002
2013-07-31 15:54:52  7088 83264 INF  12898 ENVIRON  Current user id 1001 effective id 1001
2013-07-31 15:54:52  7088 83264 INF  12898 ENVIRON  Current group id 1002 effective id 1002
2013-07-31 15:54:52  7088 83264 INF  12898 ENVIRON   cpu time unlimited
2013-07-31 15:54:52  7088 83264 INF  12898 ENVIRON   number of processes 96101
2013-07-31 15:54:52  7088 83264 INF  12898 ENVIRON   number of open files 32800
2013-07-31 15:54:52  7088 83264 INF  12898 ENVIRON   core size 0 KBytes
2013-07-31 15:54:52  7088 83264 INF  12898 ENVIRON   file size unlimited
2013-07-31 15:54:52  7088 83264 INF  12898 ENVIRON   heap memory size unlimited
2013-07-31 15:54:52  7088 83264 INF  12898 ENVIRON   stack memory size 8192 KBytes
2013-07-31 15:54:52  7088 83264 INF  12898 ENVIRON   lockable memory size 64 KBytes
2013-07-31 15:54:52  7088 83264 INF  12898 ENVIRON   virtual memory size unlimited
2013-07-31 15:54:52  7088 83264 INF  12898 ENVIRON   resident set size unlimited
2013-07-31 15:54:52  7088 83264 INF  12898 ENVIRON  Resource limit dump completed
2013-07-31 15:54:52  7088 83264 INF  12898 ENVIRON  Environment dump start
2013-07-31 15:54:52  7088 83264 INF  12898 ENVIRON   DBROOT=/sapdb/WB9/db
2013-07-31 15:54:52  7088 83264 INF  12898 ENVIRON   IFS=

2013-07-31 15:54:52  7088 83264 INF  12898 ENVIRON   LOGNAME=sdb
2013-07-31 15:54:52  7088 83264 INF  12898 ENVIRON   PATH=/bin:/usr/bin
2013-07-31 15:54:52  7088 83264 INF  12898 ENVIRON   TMP=/var/tmp
2013-07-31 15:54:52  7088 83264 INF  12898 ENVIRON   TMPDIR=/var/tmp
2013-07-31 15:54:52  7088 83264 INF  12898 ENVIRON   USER=sdb
2013-07-31 15:54:52  7088 83264 INF  12898 ENVIRON  Environment dump completed
---------------------------- end of startup part ----------------------------
2013-08-03 12:28:41 19120 83264 ERR  11934 XSERVER  Connect request: server rejected connection
2013-08-03 12:33:40 19196 83264 ERR  11934 XSERVER  Connect request: server rejected connection
```

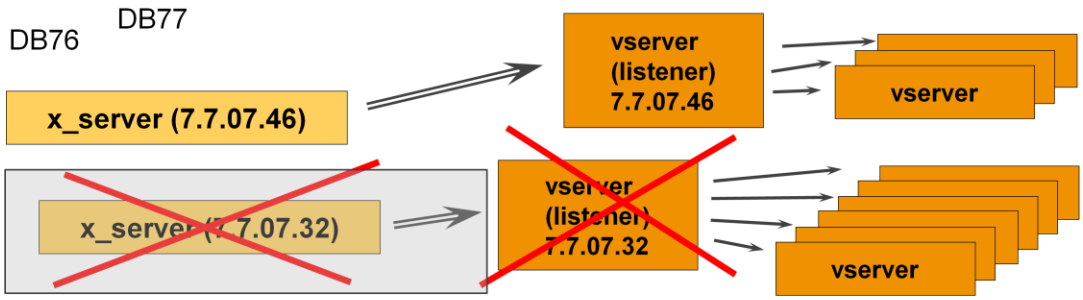
Errors in x_server logfile

```
[...]
2013-03-26 16:07:41  7333 27520 ERR -11987 COMMUNIC kernel released connection!
2013-03-26 16:07:41  7327 27520 ERR -11987 COMMUNIC kernel released connection!
2013-03-26 19:48:39  6851 27520 ERR  11926 XSERVER  Receive packet, Ref:5 - socket recv error (110:Connection timed out
2013-03-26 20:57:48  5473 27520 ERR  11926 XSERVER  Receive packet, Ref:5 - socket recv error (110:Connection timed out
2013-03-26 20:57:48  6992 27520 ERR  11926 XSERVER  Receive packet, Ref:5 - socket recv error (110:Connection timed out
2013-04-04 09:55:00 21456 27520 ERR  11379 CONNECT  Error getting TCP/IP host by address: '147.204.248.5.'
2013-04-04 09:55:14 21456 27520 ERR  11379 CONNECT  Error getting TCP/IP host by address: '147.204.248.5.'
2013-04-04 09:56:27 21456 27520 ERR  11379 CONNECT  Error getting TCP/IP host by address: '147.204.248.5.'
2013-04-04 09:57:27 21456 27520 ERR  11379 CONNECT  Error getting TCP/IP host by address: '147.204.248.5.'
2013-04-04 09:57:30 21456 27520 ERR  11379 CONNECT  Error getting TCP/IP host by address: '147.204.248.5.'
2013-04-04 10:47:03 21456 27520 ERR  11379 CONNECT  Error getting TCP/IP host by address: '147.204.248.5.'
2013-04-04 20:06:31 13581 27520 ERR  11926 XSERVER  Receive packet, Ref:5 - socket recv error (110:Connection timed out
2013-04-04 20:48:39 13616 27520 ERR  11926 XSERVER  Receive packet, Ref:5 - socket recv error (110:Connection timed out
2013-04-04 20:48:46 13599 27520 ERR  11926 XSERVER  Receive packet, Ref:5 - socket recv error (110:Connection timed out
2013-04-04 20:48:47 13614 27520 ERR  11926 XSERVER  Receive packet, Ref:5 - socket recv error (110:Connection timed out
2013-04-04 20:58:19 14302 27520 ERR  11926 XSERVER  Receive packet, Ref:5 - socket recv error (110:Connection timed out
2013-04-08 09:26:19 21456 27520 ERR  11379 CONNECT  Error getting TCP/IP host by address: '147.204.201.87.'
2013-04-08 11:34:06 21456 27520 ERR  11379 CONNECT  Error getting TCP/IP host by address: '147.204.201.87.,
2013-04-08 11:34:11 21456 27520 ERR  11379 CONNECT  Error getting TCP/IP host by address: '147.204.201.87.'
2013-04-09 09:18:37 23103 27520 ERR -11987 COMMUNIC kernel released connection!
2013-04-10 19:29:59 23111 27520 ERR  11926 XSERVER  Receive packet, Ref:5 - socket recv error (110:Connection timed out
2013-04-10 19:30:00 23117 27520 ERR  11926 XSERVER  Receive packet, Ref:5 - socket recv error (110:Connection timed out
2013-04-12 15:16:34 24448 27520 ERR  11934 XSERVER  Connect request: server rejected connection
2013-04-12 15:23:55 24558 27520 ERR  11934 XSERVER  Connect request: server rejected connection
2013-04-12 16:14:46 25393 27520 ERR  11934 XSERVER  Connect request: server rejected connection
2013-04-12 16:28:47 25683 27520 ERR  11934 XSERVER  Connect request: server rejected connection
2013-04-12 16:28:47 25685 27520 ERR  11934 XSERVER  Connect request: server rejected connection
2013-04-16 13:46:47 21456 27520 ERR  11379 CONNECT  Error getting TCP/IP host by address: '147.204.192.39.'
2013-04-16 13:46:48 21456 27520 ERR  11379 CONNECT  Error getting TCP/IP host by address: '147.204.192.39.'
2013-04-16 13:46:50 21456 27520 ERR  11379 CONNECT  Error getting TCP/IP host by address: '147.204.192.39.'
2013-04-16 13:46:51 21456 27520 ERR  11379 CONNECT  Error getting TCP/IP host by address: '147.204.192.39.'
2013-04-16 13:47:22  5674 27520 ERR -11987 COMMUNIC kernel released connection!
2013-04-16 13:47:22  5668 27520 ERR -11987 COMMUNIC kernel released connection!
2013-04-16 13:47:48 21456 27520 ERR  11379 CONNECT  Error getting TCP/IP host by address: '147.204.192.39.'
2013-04-16 13:47:51 21456 27520 ERR  11379 CONNECT  Error getting TCP/IP host by address: '147.204.192.39.'
2013-04-16 19:59:06  8095 27520 ERR  11926 XSERVER  Receive packet, Ref:5 - socket recv error (110:Connection timed out
2013-04-16 19:59:08  8101 27520 ERR  11926 XSERVER  Receive packet, Ref:5 - socket recv error (110:Connection timed out
2013-04-22 13:47:41  3081 27520 ERR  11934 XSERVER  Connect request: server rejected connection
```

Upgrade of a MaxDB version (Non-isolated installation)

As the x_server has to run with the highest version on the server it has to be exchanged if a newer version is installed.

Can be exchanged „on the fly“, i.e. not all connections to running databases have to be stopped (UNIX only).



Upgrade of DB77 to a higher patch version: 7.7.07.46

In the shown example DB77 is updated to a higher patch level within version 7.7.07. As this is the installation on the server with the highest version number the x_server has to be updated, too. Under UNIX the DB76 can continue operation when the software is exchanged. A copy of the executable is generated (in the background). Under Windows all databases on the server have to be stopped to perform the software update.

If a software update for database DB76 to a higher 7.6 is required the x_server software will not be affected.

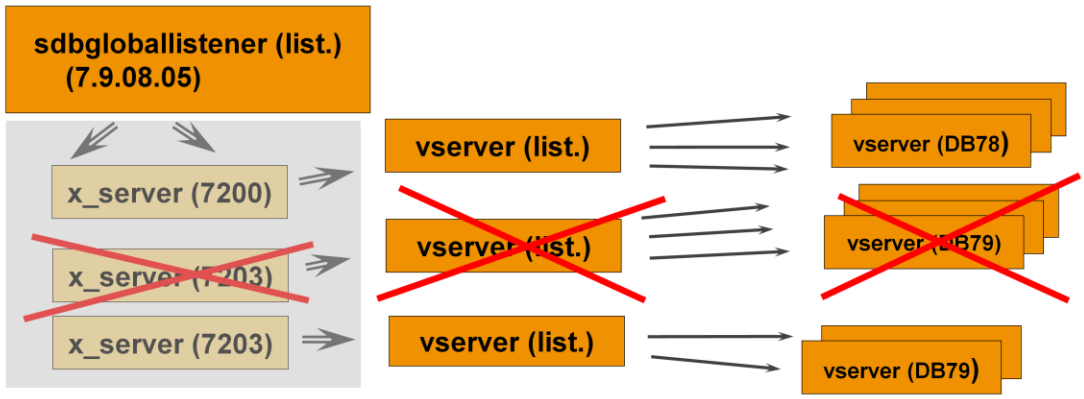
Single steps in more detail (for upgrade to a higher 7.7.07 patch):

- DB77 is stopped
- vserver (listener) and vserver processes for DB76 are still running
- during SDPUPD for DB77 a copy of vserver (listener) is generated
- the executable x_server of version 7.7.07.32 is replaced by 7.7.07.46
- when x_server (7.7.07.46) is started a new vserver (listener) has to be created
- during the short time period of the exchange also DB76 does not accept new requests for remote login

Upgrade of a MaxDB version (isolated installation)

An upgrade of a database within an isolated installation will not affect the other databases on the server.

It is not necessary to exchange the globallistener to the highest version.



Upgrade of DB79 to a higher patch version: 7.9.08.11

In this example the database DB79 is upgraded from 7.9.08.05 to 7.9.08.11. When the database is stopped all vserver processes forked by the vserver(listener) that had been started by x_server of port 7203 are stopped. All other databases are not affected. It is not necessary to stop the globallistener to bring it to the highest version.

Single steps in more detail:

- DB79 ís stopped
- the vserver (listener) of DB79 and all processes forked by it are stopped
- the vserver (listener) of DB78 continues operation
- during SDBUPD the x_server (7203) executable is exchanged
- when the software is installed the x_server of port 7203 starts again the vserver (listener) for DB79
- the sdbgloballistener itself is not affected

At rare intervals there might be logical breaks or important error corrections within the sdbgloballistener. If it should be necessary to exchange the sdbgloballistener the administrator will be informed by the installation tools.

The update of the global listener can be done separately without affecting the connections that are already established for the different databases. As the sdbgloballistener (listener) is still running (as a copy) also new connections are possible.

There will be a short time interval when the sdbgloballistener executable stops the old sdbgloballistener(listener) process and starts the new one with the upgraded version in which no new connections are accepted.

Questions

SAP® MaxDB™ x_server

Thank You!
Bye, Bye – And Remember Next Session

Feedback and further information:
http://www.sdn.sap.com/irj/sdn/maxdb

Next Session: 17.09.2013
SAP® MaxDB™ dbmserver

SAP® MaxDB™ – Expert Sessions Learning Map (1)

SAP® MaxDB ™ Features	SAP® MaxDB ™ Administration	SAP® MaxDB ™ Problem Analysis
Session 1: Low TCO with the SAP MaxDB Database	Session 2: Basic Administration with Database Studio	Session 5: SAP MaxDB Data Integrity
Session 6: New Features in SAP MaxDB Version 7.7	Session 3: CCMS Integration into the SAP System	Session 14: SAP MaxDB Tracing
Session 8: New Features in SAP MaxDB Version 7.8	Session 11: SAP MaxDB Backup and Recovery	Session 12: Analysis of SQL Locking Situations
	Session 13: Third-Party Backup Tools	
	Session 19: SAP® MaxDB™ Kernel Parameter Handling	

SAP® MaxDB ™ Installation/Upgrade
Session 7: SAP MaxDB Software Update Basics

All Expert Sessions (recording and slides) are available for download
http://maxdb.sap.com/training/

SAP® MaxDB™ – Expert Sessions Learning Map (2)

SAP® MaxDB™ Architecture	SAP® MaxDB ™ Performance
Session 18: Introduction MaxDB Database Architecture	Session 4: Performance Optimization with SAP MaxDB
Session 15: SAP MaxDB No-Reorganization Principle	Session 9: SAP MaxDB Optimized for SAP BW
Session 17: SAP MaxDB Shadow Page Algorithm	Session 16: SAP MaxDB SQL Query Optimization (Part 1)
Session 12: Analysis of SQL Locking Situations	Session 16: SAP MaxDB SQL Query Optimization (Part 2)
Session 10: SAP MaxDB Logging	
Session 20: SAP MaxDB Remote SQL Server	
Session 21: SAP MaxDB DBM Server	

All Expert Sessions (recording and slides) are available for download
http://maxdb.sap.com/training/

Thank you

Contact information:

Heike Gursch
IMS NW MaxDB / liveCache
Heike.Gursch@sap.com